NUSC Technical Document 6317
30 March 1981

A Gain Step Companding (COMpressing ExPANDING) Analog-To-Digital Converter

Gerald L. Assard
Surface Ship Sonar Department

Naval Underwater Systems Center
Newport, Rhode Island / New London, Connecticut

Preface

This document was prepared under the Ocean Measurements and Array Technology (OMAT) Program portion of the SEAGUARD Program sponsored by the Defense Advanced Research Projects Agency (ARPA Order No. 2976), Program Manager, V. Simmons, Tactical Technology Office, NUSC Project No. A69600, Program Manager, R. F. LaPlante (Code 33492).

The author gratefully acknowledges the contribution of Dr. Jude Franklin of MAR, Incorporated, for his assistance in the development, preparation, and editing of this technical document.

Reviewed and Approved: 30 March 1981

D. Walters
Head, Surface Ship Sonar Department

The author of this document is located at the
New London Laboratory, Naval Underwater Systems Center
New London, Connecticut 06320

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
NUSC-TD-6317	AD-A099 381	

4. TITLE (and Subtitle)

A GAIN STEP COMPANDING (COMpressing ExPANDING) ANALOG-TO-DIGITAL CONVERTER

5. TYPE OF REPORT & PERIOD COVERED

Technical document

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

Gerald L. Assard

8. CONTRACT OR GRANT NUMBER(s)

ARPA Order-2976

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Naval Underwater Systems Center
New London Laboratory
New London, CT 06320

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

Ocean Measurements & Array Technology (OMAT) SEAGUARD No. A-696-00 ARPA Ord#2976

11. CONTROLLING OFFICE NAME AND ADDRESS

Defense Advanced Research Projects Agency
1400 Wilson Blvd.
Arlington, VA 22209

12. REPORT DATE

30 Mar 81

13. NUMBER OF PAGES

20 (12) 23

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)

UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Analog-to-Digital Converter
Auto-Ranging
Binary-to-Exponent-Mantissa Conversion
Decimal-to-Binary Conversion
Digital Telemetry System
Gain Step Companding (COMpressing ExPANDING)
Mantissa

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A novel scheme for analog-to-digital conversion is described. This scheme provides a dynamic range in excess of 120 dB yet requires an output of only 9 bits. These 9 bits are utilized to transmit a companded digital word format including sign information. Use of this companded output format results in significant transmit bandwidth reduction compared with transmit bandwidths required for conventional linear bit formats. This companding results in a worst case error of 3.125%.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

405918

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Section 1

BACKGROUND

Digital telemetry systems are required in most towed underwater sonars in order to multiplex and digitze a large number of acoustic and nonacoustic sensors. Usually the telemetry must be capable of transmitting its signal in a reliable manner to a cable/array system 1.6093 to 3.2186 kilometers (1 to 2 miles) in length. The cable diameter is minimized to reduce hydrodynamic drag, and this limits the overall telemetry bandwidth. In addition, the telemetry system must be capable of operating over a very large acoustic dynamic range and increasing individual channel bandwidths. These requirements,

- large number of sensors,
- limited cable bandwidth,
- high dynamic range, and
- wide acoustic sensor bandwidths,

are often conflicting; and they oblidge the designers of towed acoustic arrays to seek out novel and innovative schemes to reduce telemetry bandwidths, yet capable of meeting the large dynamic range requirements. This document describes a scheme that can be used over a dynamic range in excess of 120 dB that requires only 9 bits per channel. Normally a system that operates over a 120 dB dynamic range would need 20 bits per channel. The reduction from 20 to 9 bits per channel is a savings of over 2 to 1 in bandwidth reduction. Thus, this system is beneficial in that it meets the large dynamic range requirements, yet reduces the telemetry bandwidth requirements.

There are many applications in which this scheme could be incorporated, for example:

1. loud targets at close range,
2. intercept of communication signals,
3. seismic research (for which dynamic range requirements are very high),

4. radiated noise measurements in a TUMS (Towed Underwater Measurement System),
5. acquisition of transient radiated signals,
6. mobile sonar used in quiet and also noisy environments,
7. bistatic sonar used with towed arrays,
8. multiple targets in a task force environment or a direct support role for SSN's,
9. towed and hull-mounted arrays,
10. high-speed torpedo targets.

Section 2

TECHNICAL DISCUSSION

Figure 2-1 describes the technique used to implement the 19-bit plus sign conversion into an 8-bit (4-bit mantissa and 4-bit exponent) plus sign output. The signal conditioner (see (1) in figure 2-1) is used to prewhiten the input signal such that it is nearly a flat spectrum going into the gain control units (2, 3, 4, 5, 6, 7). The analog portion changes the level of the signal in order to maintain the signal within an amplitude window, as described below. The analog-to-digital converter (ADC) (8) outputs 11 bits plus sign to a code converter that also has an input from the gain control (7) unit. These two inputs are converted into an 8-bit plus sign code by the code converter (9).

2.1 CIRCUIT DESCRIPTION (ANALOG PART)

The signal output of the signal conditioner is fed to the input of the digitally switched gain amplifier (see (2) in figure 2-1). The gain of the amplifier is made to decrease in discrete steps as the input voltage to it increases, as follows.

For low-level input signals, the amplifier is switched to its maximum gain of 48 dB. The output of the amplifier is fed to the AC/DC converter (see (3) in figure 2-1) whose output is fed into the low-pass filter (4). The low-pass filter time constant is an order of magnitude greater than the period of the lowest frequency of interest, so that the filter output is a DC signal proportional to the mean value of the signal at the amplifier output.

The low-pass filter DC output is fed to the upper threshold detector (see (5) in figure 2-1) and to the lower threshold detector (6). If the

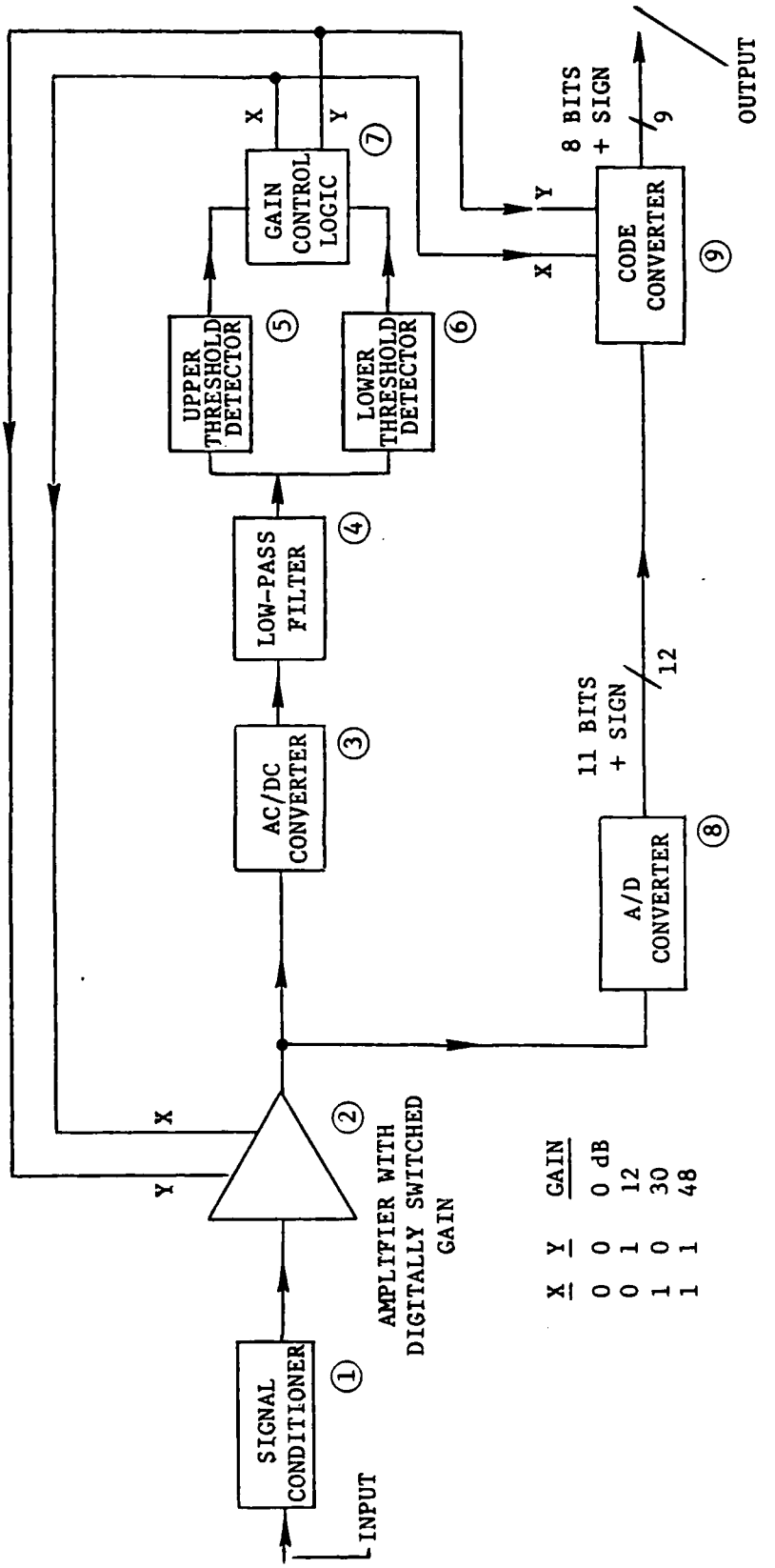

Figure 2-1. Gain Step 19-Bit Plus Sign ADC Block Diagram

input signal is slowly increasing in amplitude from 0, the amplifier gain will remain at 48 dB until the DC voltage out of the low-pass filter reaches the upper threshold value. At this point, the upper threshold detector (5) sends a signal to the gain control logic (7) commanding a decrease in gain to 30 dB. The amplifier output level and the DC voltage out of the low-pass filter will drop by 18 dB.

The lower threshold level is set 30 dB below the upper threshold level, so the lower threshold is not exceeded when the gain drops 18 dB. As this input signal continues to increase, the upper threshold is again reached and a decrease in gain to 12 dB is commanded. Further increase in the input signal level causes a gain decrease to 0 dB, which is the minimum gain of the amplifier. As the input signal level decreases from its maximum level, the amplifier gain will be increased as described above, only in this case it will involve the lower threshold detector (6).

Figure 2-2 shows how the mean level of the output voltage of the digitally switched gain amplifier varies as the mean level of the input to the amplifier first increases, and then decreases.

2.2 CIRCUIT DESCRIPTION (DIGITAL PART)

The digital portion of this unit includes the 11-bit plus sign ADC and the code converter. In effect, the combination of the gain control and the ADC can be viewed as an equivalent 19-bit plus sign ADC. This equivalent 20-bit ADC is described further in this section. Figure 2-3 indicates the way this 19-bit plus sign ADC is synthesized from the 11-bit plus sign ADC and the two gain control bits. When the gain is maximum (indicated by a code of 11), the 11-bit output is in the least significant bits of the 20-bit synthesized ADC (0 through 10). As the analog signal increases and the gain drops to 10, the 11-bit ADC output is in effect moved within the 20-bit synthesized ADC. This relative motion of the 11-bit output is continued as the gain is changed, as indicated in figure 2-3.

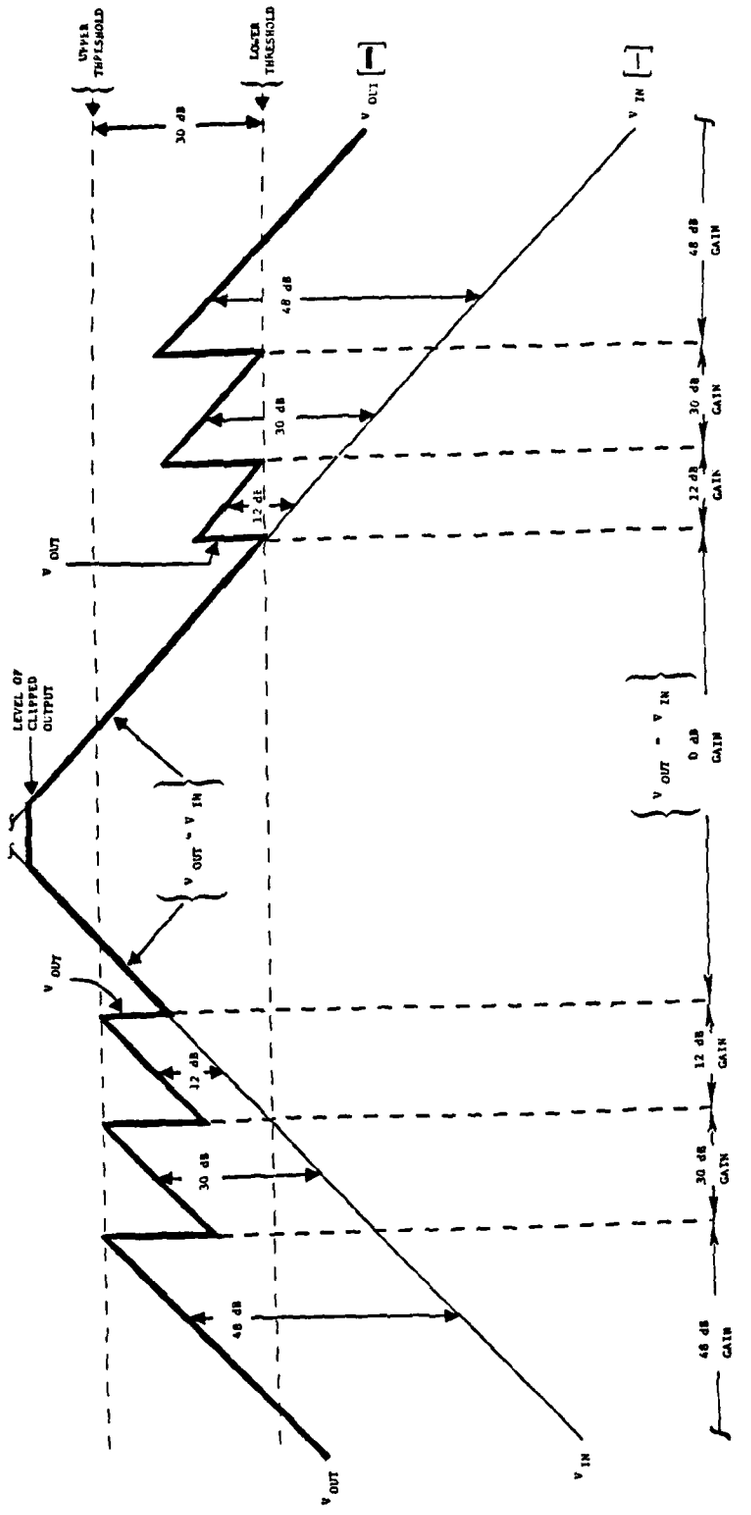

Figure 2-2. Auto-Ranging Amplifier with Gains of 0, 12, 30, and 48 dB and a 30-dB Threshold Window

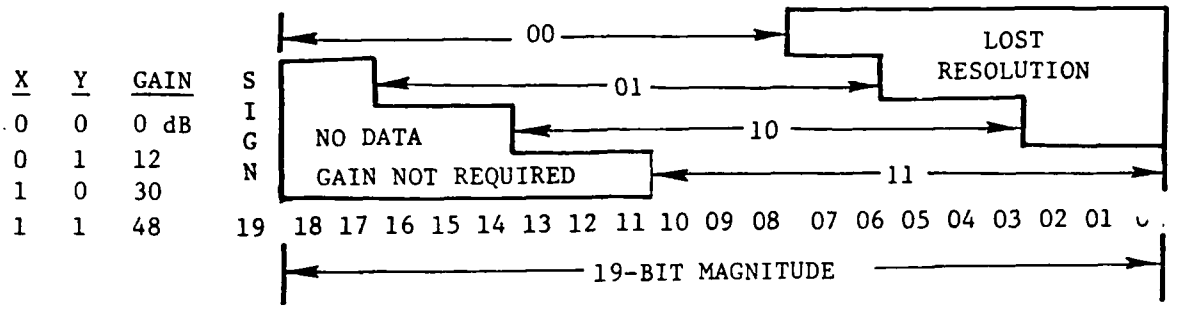

Figure 2-3. Gain Step 20-Bit ADC

The companding ADC is a method of reducing the bit requirements of a linear multibit ADC.[1] In conventional notation, a decimal number can be represented as a binary number using a decimal-to-binary conversion table such as table 2-1. Each decimal number can be uniquely represented, using a conversion table, by a set of bits (binary digits), and conversely each set of bits represents only one decimal number. But, as noted near the bottom of table 2-1, as the magnitude of the decimal numbers increases, the number of binary digits required increases tremendously.

An alternate method of representing decimal numbers with a slight loss of accuracy is a scheme using an exponent and mantissa. The exponent denotes the position of the most significant nonzero bit. Those bits following the first nonzero bit are used in generating the mantissa. The exponent-mantissa method of representing a 19-bit binary number is shown in table 2-2.

---

[1] Transmission Systems for Communications, Bell Telephone Laboratories, Inc., Western Electric Company, Inc., Winston-Salem, NC, 1971, pp. 570-583.

Table 2-1. Sample Decimal-to-Binary Conversion Table

Decimal Number	Binary Equivalent
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111
⋮	⋮
31	11111
⋮	⋮
63	111111
⋮	⋮
127	1111111
⋮	⋮
255	11111111
⋮	⋮
511	111111111
⋮	⋮
1023	1111111111
⋮	⋮
2047	11111111111
⋮	⋮
4095	111111111111

Table 2-2. Binary-to-Exponent-Mantissa Conversion Table

Decimal Number	19-Bit Binary Number																			Exponent	Mantissa	
	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	0000	
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0000	0001	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0000	0010	
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0000	0011	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0000	0100	
:										:										:	:	Zone 1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0000	1111	
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0001	0000	
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0001	0001	
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0001	0010	
:										:										:	:	
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0001	1111	
32	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0010	0000	
33	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0010	0000	
34	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0010	0001	
35	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0010	0001	
36	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0010	0010	
37	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0010	0010	
:										:										:	:	Zone 2
62	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0010	1111	
63	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0010	1111	
														Exp. Gen.	Mantissa Generation				Lost Bits			

Table 2-2. (Cont'd) Binary-to-Exponent-Mantissa Conversion Table

Decimal Number	19-Bit Binary Number																			Exponent	Mantissa	
	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
64	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0011	0000	
65	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0011	0000	
66	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0011	0000	
67	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0011	0000	Zone 3
68	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0011	0001	
:											:									:	:	
127	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0011	1111	
128	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0100	0000	
129	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0100	0000	
130	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0100	0000	
131	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0100	0000	
132	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0100	0000	
133	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0100	0000	
134	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0100	0000	
135	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	0100	0000	
136	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0100	0001	
:											:									:	:	
255	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0100	1111	Zone 4
											Exponent Generation		Mantissa Generation				Lost Bits					

Table 2-2. (Cont'd) Binary-to-Exponent-Mantissa Conversion Table

Decimal Number	19-Bit Binary Number																			Exponent	Mantissa	
	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
256	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0101	0000	
:																				:	:	Zone 5
511	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0101	1111	
512	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0110	0000	
:										:										:	:	Zone 6
1023	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0110	1111	
1024	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0111	0000	
:										:										:	:	Zone 7
2047	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0111	1111	
2048	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1000	0000	
:										:										:	:	Zone 8
4095	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1000	1111	
4096	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1001	0000	
:										:										:	:	Zone 9
8191	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1001	1111	
8192	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1010	0000	
:										:										:	:	Zone 10
16383	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1010	1111	
						Exponent Generation				Mantissa Generation									Lost Bits			

Table 2-2. (Cont'd) Binary-to-Exponent-Mantissa Conversion Table

Decimal Number	19 Bit-Binary Number																			Exponent	Mantissa	
	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
16384	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1011	0000	
:										:										:	:	Zone 11
32767	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1011	1111	
32768	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1100	0000	
:										:										:	:	Zone 12
65535	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1100	1111	
65536	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1101	0000	
:										:										:	:	Zone 13
131071	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1101	1111	
131072	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1110	0000	
:										:										:	:	Zone 14
262143	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1110	1111	
262144	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1111	0000	
:										:										:	:	Zone 15
524287	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1111	1111	

Exponent Generation

Mantissa Generation

Lost Bits

In table 2-2, a 4-bit exponent and a 4-bit mantissa represent a 19-bit binary number. The method of converting a binary number to exponent-mantissa notation is as follows:

1. Start at the extreme left bit, Bit 18.
2. Is this bit a "1"?
   - A. If yes, go to Line 3.
   - B. If no, move to the next bit on the right, and go back to Line 2.
3. Compute exponent:
   Exponent = (bit no. -3). If expression is negative, Exponent = 0.
4. Create mantissa by using the next four bits exactly as they appear in the binary number.

The above sequence will represent without error all the binary numbers appearing in Zone 1 of table 2-2. Each exponent-mantissa pair uniquely denotes one binary number.

Upon inspection of the exponent-mantissa pairs in Zone 2, we see that redundant mantissas can be found for different binary numbers. Two binary numbers are represented by the same exponent-mantissa pair. In Zone 3, four binary numbers are represented by the same exponent-mantissa pair. The situation worsens in subsequent zones. When attempting to use an exponent-mantissa pair to recover the original binary number, the occurrence of redundant mantissas implies that there is more than one binary number possible. This is called conversion error.

The conversion error can be computed using an exponent-mantissa pair. As shown below, the same exponent-mantissa pair represents decimal numbers 32 and 33:

Exponent	Mantissa	Binary Number	Decimal Number
0010	0000	0000000000000100000	32
0010	0000	0000000000000100001	33

Employing the exponent and mantissa, reconstruct the original binary number as follows:

1. Start with a 19-bit field

18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

2. Use the exponent to determine the position of the first non-zero bit: Exp = 2 = (Bit No. -3) Bit No. = 5

18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1					

3. Use the mantissa to fill in the next four bits:

18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	

4. The exponent and mantissa do not specify the status of the remaining bit, Bit 0. In order to complete the 19-bit field, assign a value of 1 to Bit 0.

18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

5. The binary number recovered from the exponent and mantissa represents decimal 33. If the original decimal number was 33, there is no conversion error. If the original number was 32, compute conversion error as follows:

$$\text{Conversion Error} = \left| \frac{\text{Original Number - Recovered Number}}{\text{Original Number}} \right| \times 100$$

$$= \left| \frac{32 - 33}{32} \right| \times 100 = 3.125\% \text{ Error.}$$

Moving down to Zone 3, compute conversion error for the numbers below:

Exponent	Mantissa	Binary Number	Decimal Number
0011	0001	0000000000001000100	68
0011	0001	0000000000001000101	69
0011	0001	0000000000001000110	70
0011	0001	0000000000001000111	71

The mantissa and exponent will generate

0000000000010001XX

using the above procedure, with the X's denoting the ambiguous bit positions. By forcing the first ambiguous bit to one and the remaining bit to zero, the resulting number is

0000000000001000110

or 70.

The percentage of error for all the possible original values is computed below:

$$68: \left|\frac{70 - 68}{68}\right| \times 100 = 2.94\%$$

$$69: \left|\frac{70 - 69}{69}\right| \times 100 = 1.45\%$$

$$70: \left|\frac{70 - 70}{70}\right| \times 100 = 0\%$$

$$71: \left|\frac{70 - 71}{71}\right| \times 100 = 1.41\%$$

A typical error computation for Zone 4 appears below:

Exponent	Mantissa	Binary Number	Decimal Number
0100	0001	0000000000010001000	136
0100	0001	0000000000010001001	137
0100	0001	0000000000010001010	138
0100	0001	0000000000010001011	139
0100	0001	0000000000010001100	140
0100	0001	0000000000010001101	141
0100	0001	0000000000010001110	142
0100	0001	0000000000010001111	143

1. Number generated by exponent and mantissa of 0100 0001:
   0000000000010001XXX
2. Assign value of 1 to first X:
   0000000000010001 1XX
3. Assign zero to remaining X's:
   $0000000000010001100_2 = 140_{10}$

4. Compute conversion error:

$$136: \left|\frac{136 - 140}{136}\right| \times 100 = 2.94\%$$

$$137: \left|\frac{137 - 140}{137}\right| \times 100 = 2.19\%$$

$$138: \left|\frac{138 - 140}{138}\right| \times 100 = 1.45\%$$

$$139: \left|\frac{139 - 140}{139}\right| \times 100 = 0.72\%$$

$$140: \left|\frac{140 - 140}{140}\right| \times 100 = 0\%$$

$$141: \left|\frac{141 - 140}{141}\right| \times 100 = 0.71\%$$

$$142: \left|\frac{142 - 140}{142}\right| \times 100 = 1.41\%$$

$$143: \left|\frac{143 - 140}{143}\right| \times 100 = 2.10\%$$

For example, to illustrate maximum error, the following procedure is used to transfer 128_{10}:

$128_{10} = 10000000_2$ (↑ 2^{exp}) — Data to be Transmitted

$132_{10} = 10000100_2$ (↑ 2^{exp-5}) — Data Reconstructed

$132_{10} = 00000000000100001000_2$ — 19-Bit Data Output Representation,

where 2^{exp-5} is added in the data reconstruction to minimize the worst case error. The equation describing the minimized maximum error is

For any valued exp:

$$\text{Maximum Error} = \frac{(2^{exp} + 2^{exp-5}) - (2^{exp})}{2^{exp}}$$

$$= \frac{(2^{exp})\ (2^{-5})}{2^{exp}}$$

$$= 2^{-5}$$

$$= \frac{1}{32} \text{ or } 3.125\%.$$

As can be seen from the above discussion, not only is it possible to represent a 19-bit binary number with a 4-bit exponent and a 4-bit mantissa, but also it can be done with a maximum error of 3.125%. This is the scheme that is used to reduce the bit requirements of a linear multibit ADC.

Section 3

SUMMARY AND CONCLUSIONS

This document describes a novel scheme that can meet the stringent dynamic range requirements and, yet, reduce the required telemetry bandwidth requirements. Although the analog-to-digital converter described has a dynamic range in excess of 120 dB, it requires only 9 bits to transmit a companded format of sign (1 bit), exponent (4 bits), and mantissa (4 bits). These new terms are described in this document. The worst-case error was less than 3.125%. Several applications of this analog-to-digital converter are also discussed.

INITIAL DISTRIBUTION LIST

Addressee	No. of Copies
ASN (RE&S)	1
OUSDR&E (Res. & Adv. Tech)	1
ONR, ONR-480, -100, -212, -222, -468	5
CNO, OP-952, -955, -981H, -951, -951D, 951E	6
CNM, MAT-08T21, SPO PM-2 (2)	3
NAVELECSYSCOM, PME-124 (3), ELEX 320 (3)	6
NAVSEASYSCOM, SEA-61R, -63R	2
NAVAIRDEVCEN	1
NOSC (R. Smith)	1
DAVWPNCEN	1
NAVPGSCOL	1
DTIC	12
DARPA	5
NORDA (R. Swenson, R. Martin, R. Rumpf, R. Gardner)	4
NEL (B. Adams)	2
NAVPGSCOL	1
NAVSURFWPNCEN	1
DWTNSRDC ANNA	1
DWTNSRDC CARD	1